NYO-1480-82

Courant Institute of
Mathematical Sciences

AEC Computing and Applied Mathematics Center

The Numerical Solution of the Navier-Stokes Equations for an Incompressible Fluid

Alexandre Joël Chorin

AEC Research and Development Report

Mathematics
November 1967

New York University

UNCLASSIFIED

AEC Computing and Applied Mathematics Center
Courant Institute of Mathematical Sciences
New York University

Mathematics NYO-1480-82

THE NUMERICAL SOLUTION OF THE NAVIER-STOKES EQUATIONS FOR AN INCOMPRESSIBLE FLUID

Alexandre Joël Chorin

Contract No. AT(30-1)-1480

UNCLASSIFIED

Abstract

A finite difference method for solving the time-dependent Navier-Stokes equations for an incompressible fluid is introduced. This method uses the primitive variables, i.e. the velocities and the pressure, and is equally applicable to problems in two and three space dimensions. Test problems are solved, and an application to a three dimensional convection problem is presented.

Introduction. The equations of motion of an incompressible fluid are

$$\partial_t u_i + u_j \partial_j u_i = - \frac{1}{\rho_o} \partial_i p + \nu \nabla^2 u_i + E_i \qquad (\nabla^2 \equiv \sum_j \partial_j^2)$$

$$\partial_j u_j = 0$$

where u_i are the velocity components, p is the pressure, ρ_o is the density, E_i are the components of the external forces per unit mass, ν is the coefficient of kinematic viscosity, t is the time, and the indices i,j refer to the space coordinates x_i, x_j, i, j = 1,2,3. ∂_i denotes differentiation with respect to x_i, and ∂_t differentiation with respect to the time t. The summation convention is used in writing the equations.

We write

$$u_i' = \frac{u_i}{U} \qquad x_i' = \frac{x_i}{d} \qquad p' = (\frac{d}{\rho_o \nu U})p$$

$$E_i' = (\frac{\nu U}{d^2})E_i \qquad t' = (\frac{\nu}{d^2})t$$

where U is a reference velocity, and d a reference length. We then drop the primes. The equations become

$$\partial_t u_i + R u_j \partial_j u_i = - \partial_i p + \nabla^2 u_i + E_i \qquad (1)$$

$$\partial_j u_j = 0 \qquad (2)$$

where $R = Ud/\nu$ is the Reynolds number. It is our purpose

to present a procedure for solving these equations by finite differences in some bounded domain $\mathcal{D}$, in either two or three space variables. The distinguishing feature of this method is the use of equations (1) and (2) rather than higher order derived equations. This makes it possible to satisfy all boundary conditions and to achieve adequate computational efficiency even in problems involving three space dimensions and time. The author is not aware of any other method for which these claims can be made.

Principle of the method. Let u_i, p denote the solution of (1) - (2) as well as its discrete approximation, and let $Du = 0$ be a difference approximation to $\partial_j u_j = 0$. In general we shall allow Du to take different forms in the interior of the domain $\mathcal{D}$ and on its boundary; at the boundary we may wish to use higher order one-sided differences so as to preserve accuracy of some given order. It is assumed that at time $t = n\Delta t$ velocity and pressure fields u_i^n, p^n are given satisfying $Du^n = 0$. The task at hand is to evaluate u_i^{n+1}, p^{n+1} using equation (1), so that $Du^{n+1} = 0$.

Auxiliary fields u_i^{aux}, are first computed through

$$u_i^{aux} = u_i^n + \Delta t \, F_i u \tag{3}$$

where $F_i u$ approximates

$$- Ru_j\partial_j u_i + \nabla^2 u_i + E_i \qquad \nabla^2 \equiv \sum \partial_j^2$$

$F_i u$ may depend on u_i^n, u_i^{aux}, and intermediate fields, say u_i^*, u_i^{**}. In the evaluation of u_i^{aux} equation (2) and the pressure term in equation (1) are not taken into account. An iteration procedure is now introduced for the purpose of finding u_i^{n+1} inside $\mathcal{D}$ and p^{n+1} in $\mathcal{D}$ and on its boundary, by setting

$$p^{n+1,1} = p^n \tag{4a}$$

$$u_i^{n+1,m+1} = u_i^{aux} - \Delta t G_i^m p \qquad m \geq 1 \tag{4b}$$

$$p^{n+1,m+1} = p^{n+1,m} - \lambda D u^{n+1,m+1} \qquad m \geq 1 \tag{4c}$$

where λ is a parameter, the quantities $u_i^{n+1,m}$ and $p^{n+1,m}$ are successive approximations to u_i^{n+1}, p^{n+1}, and $G_i^m p$ is a function of $p^{n+1,m}$ and $p^{n+1,m+1}$ which converges to a difference form of $\partial_i p^{n+1}$ as $|p^{n+1,m+1} - p^{n+1,m}|$ tends to zero. The form of $G_i^m p$ is crucial for the convergence of the iterations and for the accuracy of the over-all scheme; it will be specified below.

When for some ℓ and a small predetermined constant ε

$$\max_{\mathcal{D}} |p^{n+1,\ell+1} - p^{n+1,\ell}| \leq \varepsilon \tag{5}$$

we set

$$u_i^{n+1} = u_i^{n+1,\ell+1}$$

$$p^{n+1} = p^{n+1,\ell+1}$$

If the velocity component u_i is prescribed at the boundary, $u_i^{n+1,m+1}$ in (4b) and (4c) is replaced by the given value of u_i^{n+1}. The iterations (4) insure that equation (1), including the pressure term, is satisfied inside $\mathcal{O}$, and equation (2) is satisfied in $\mathcal{O}$ and on its boundary.

The proposed method is essentially a generalization of the artificial compressibility method, introduced in [1] for the purpose of finding steady solutions of equations (1) - (2). This can be seen in the following way: Let us drop the requirement (5) and set $\ell = 1$ for all n. t then does not represent real time, since equation (2) is not satisfied and we are not following the evolution in real time of the solutions of (1) - (2). We choose for $G_i^1 p$ an expression which approximates $\partial_i p$, and set $\delta = \Delta t/\lambda$. Remembering the definition of u_i^{aux}, we see that equations (4) represent one step in an artificial time for a system approximating the equations

$$\begin{aligned} \partial_t u_i + Ru_j \partial_j u_i &= -\partial_i p + \nabla^2 u_i + E_i \\ \partial_t p + \frac{1}{\delta}\partial_j u_j &= 0 \end{aligned} \qquad (6)$$

This is the auxiliary system used in [1] to find steady solutions of equations (1) - (2). If, as the artificial time is advanced, the solution of (6) tends to a steady limit, i.e. one which does not depend on t, then this solution is a steady solution of equations (1) and (2).

Comparison with the method of Harlow and Welch. Before introducing specific scheme for evaluating u_i^{aux} and a specific representation for Du and $G_i^m p$, we shall compare our method to the only other method known to the author for solving the incompressible Navier-Stokes equations using the primitive variables. That method is due to Harlow and Welch [2], who applied it to two dimensional problems. The purpose of the comparison is to explain what the iterations (4) accomplish. It should be noted that in an important respect the method of Harlow and Welch is more ambitious then ours, having been applied to flows with a free surface, while we have thus far attempted to solve only problems for a confined fluid. Disregarding for the moment the details of the difference scheme used by Harlow and Welch, and using notations similar to those introduced in the preceding section, we can summarize their method as follows:

Let $Du = 0$ approximate $\partial_j u_j = 0$, and $G_i p$ approximate $\partial_i p$. It is assumed that at time $t = n\Delta t$ velocity fields u_i^n are given, satisfying $Du^n = 0$. Equation (2) can be approximated by

$$u_i^{n+1} = u_i^n + \Delta t L u_i^n - \Delta t Q_i u^n - \Delta t Q_i p^n + \Delta t E_i \qquad (7)$$

where Lu approximates $\nabla^2 u$, and $Q_i u$ approximates $\partial_j u_i u_j$.

Performing the operation D on (7), and assuming

$$Du^{n+1} = 0$$

we obtain the following equation

$$L' p^n = - \frac{Du^n}{\Delta t} + DLu^n - DQu^n + DE_i \qquad (8)$$

where $L' p \equiv DGp$ approximates $\nabla^2 p$. This equation is of course a difference analogue of the equation

$$\nabla^2 p = - \partial_i \partial_j u_i u_j + \partial_j E_j \qquad (9)$$

which can be obtained from equation (1) by taking its divergence. Equation (8) is solved by iteration; the boundary conditions are obtained by applying equation (1) to the fluid at the boundary. The pressure field p^n thus obtained is used in equation (7) to evaluate u_i^{n+1}; we then have $Du^{n+1} = 0$.

Our method naturally bears a resemblance to the method of Harlow and Welch. By substituting equation (4b) into (4c), which is possible except near a boundary, we obtain the equation

$$p^{n+1,m+1} - p^{n+1,m} = - \lambda Du^{aux} + \Delta t \, DG^m p$$

which, in view of the definitions of D, G_i^m and u_i^{aux}, is an iteration procedure for solving an analogue of equation (8). In this sense our method is related to Harlow and Welch's like a predictor-corrector method to a predictor method; whereas Harlow and Welch first determine p^n so that $Du^{n+1} = 0$ we make a guess at the values at u_i^{n+1}, p^{n+1}, and then correct them until the constraint $Du^{n+1} = 0$ is satisfied.

The advantage of the predictor-corrector approach is that it allows us to evaluate u_i^{aux} by implicit schemes, and thus take much larger time steps; this is of course particularly useful if three dimensional problems are to be solved.

Harlow and Welch retain the terms $\frac{1}{\Delta t} Du^n$, $DLu^n = LDu^n$ in equation (8), although it assumed that $Du^n = 0$. This allows them to reduce the stringency of the convergence criterion in the iterative solution of (8), and thus to reduce the computational effort. If we consider the definition of u_i^{aux}, we see that this feature of Harlow and Welch's method is preserved in ours in a natural way; we shall see in a later section that through an appropriate choice of $G_i^m p$ we can introduce further time-saving devices.

It is believed that the main advantage of our formulation over that of Harlow and Welch lies in the following: the system consisting of equation (1) and (2) and the one consisting of (1) and (9) are equivalent only if equation (2) is satisfied at all boundaries. In order to ensure that (2) is satisfied Harlow and Welch introduce artificial reflection principles at the boundaries; in effect, this means that the assigned boundary data are not imposed on equation (2); the solution thus obtained has therefore an uncertain relation to that of the problem to be solved. In our formulation, no such problem arises; equations (1) and (2) are consistently approximated everywhere, and the correct boundary data used.

Harlow and Welch implement their method with a difference scheme which contains many attractive features, and which could be used with our formulation. The schemes we are about to introduce appear to be much more efficient than theirs, but suitable mainly for a problem where the domain $\mathcal{D}$ has a simple shape, and where the boundary data are smooth.

Evaluation of u_i^{aux}. We shall now present schemes for evaluating u_i^{aux}, defined by (3).

In [1] we used a combined Dufort-Frankel and leap-frog scheme, in which time and the first space derivatives were approximated by centered differences, and a second derivative such as

$$\partial_1^2 u$$

was replaced by

$$\frac{1}{\Delta x_1^2}\left(u_{q+1}^n + u_{q-1}^n - u_q^{n+1} - u_q^{n-1}\right) \qquad u_q^n \equiv u(q\Delta x_1, n\Delta t)$$

This scheme is appropriate only when an asymptotic steady solution is sought. It is however inacceptably inaccurate when real time dependence is studied, unless Δt is inadmissibly small. Our reason for discussing this scheme here is the following: The Dufort-Frankel scheme is explicit and unconditionally stable; it is therefore a natural scheme to use when the non-linear terms in (1) are differenced in

"conservation-law" form, i.e. in the form

$$\partial_j(u_i u_j)$$

rather than

$$u_j \partial_j u_i \ .$$

We found that in problems in which the viscosity is not small, it is preferable to use "non-conservative" difference schemes for the non-linear terms, and avoid the Dufort-Frankel scheme.

We shall instead be using one of the following variants of the alternating-direction implicit method:

A. In two-dimensional problems we shall use a Peaceman-Rachford scheme, as proposed by Wilkes [3] in a different context. This takes the form

$$\begin{aligned} u^*_{i(q,r)} = u^n_{i(q,r)} &- R\,\frac{\Delta t}{4\Delta x_1}\,u^n_{1(q,r)}\left(u^*_{i(q+1,r)} - u^*_{i(q-1,r)}\right) \\ &- R\,\frac{\Delta t}{4\Delta x_2}\,u^n_{2(q,r)}\left(u^n_{i(q,r+1)} - u^n_{i(q,r-1)}\right) \\ &+ \frac{\Delta t}{2\Delta x_1^2}\left(u^*_{i(q+1,r)} + u^*_{i(q-1,r)} - 2u^*_{i(q,r)}\right) \\ &+ \frac{\Delta t}{2\Delta x_2^2}\left(u^n_{i(q,r+1)} + u^n_{i(q,r-1)} - 2u^n_{i(q,r)}\right) \\ &+ \frac{\Delta t}{2}\,E_i \end{aligned} \qquad (10a)$$

$$u^{aux}_{i(q,r)} = u^{*}_{i(q,r)} - R\,\frac{\Delta t}{4\Delta x_1}\,u^{*}_{1(q,r)}\left(u^{*}_{i(q+1,r)} - u^{*}_{i(q-1,r)}\right)$$

$$- R\,\frac{\Delta t}{4\Delta x_2}\,u^{*}_{2(q,r)}\left(u^{aux}_{i(q,r+1)} - u^{aux}_{i(q,r-1)}\right)$$

$$+ \frac{\Delta t}{2\Delta x_1^2}\left(u^{*}_{i(q+1,r)} + u^{*}_{i(q-1,r)} - u^{*}_{i(q,r)}\right)$$

$$+ \frac{\Delta t}{2\Delta x_2^2}\left(u^{aux}_{i(q,r+1)} + u^{aux}_{i(q,r-1)} - 2u^{aux}_{i(q,r)}\right)$$

$$+ \frac{\Delta t}{2}\,E_i$$

where u^{*}_{i} are auxiliary fields, and $u_{i(q,r)} \equiv u_i(q\Delta x_1, r\Delta x_2)$. As usual, the one-dimensional systems of algebraic equations can be solved by Gaussian elimination.

B. In two dimensional and three dimensional problems we shall use a variant of the alternating direction method analyzed by Samarskii in [4]. This takes the form

$$u^{*}_{i(q,r,s)} = u^{n}_{i(q,r,s)} - R\,\frac{\Delta t}{2\Delta x_1}\,u^{n}_{1(q,r,s)}\left(u^{*}_{i(q+1,r,s)} - u^{*}_{i(q-1,r,s)}\right)$$

$$+ \frac{\Delta t}{\Delta x_1^2}\left(u^{*}_{i(q+1,r,s)} + u^{*}_{i(q-1,r,s)} - 2u^{*}_{i(q,r,s)}\right)$$

$$u^{**}_{i(q,r,s)} = u^{*}_{i(q,r,s)} - R\,\frac{\Delta t}{2\Delta x_2}\,u^{*}_{2(q,r,s)}\left(u^{**}_{i(q,r+1,s)} - u^{**}_{i(q,r-1,s)}\right)$$

$$+ \frac{\Delta t}{\Delta x_2^2}\left(u^{**}_{i(q,r+1,s)} + u^{**}_{i(q,r-1,s)} - 2u^{**}_{i(q,r,s)}\right)$$

$$u^{aux}_{i(q,r,s)} = u^{**}_{i(q,r,s)} - R\frac{\Delta t}{2\Delta x_3} u^{**}_{3(q,r,s)}(u^{aux}_{i(q,r,s+1)} - u^{aux}_{i(q,r,s-1)})$$
$$+ \frac{\Delta t}{\Delta x_3^2}(u^{aux}_{i(q,r,s+1)} + u^{aux}_{i(q,r,s-1)} - 2u^{aux}_{i(q,r,s)})$$
$$+ \Delta t E_{i(q,r,s)}$$

$$u_{i(q,r,s)} \equiv u_i(q\Delta x_1, r\Delta x_2, s\Delta x_3)$$

$$E_{i(q,r,s)} \equiv E_i(q\Delta x_1, r\Delta x_2, s\Delta x_3)$$

u^*_i, u^{**}_i are auxiliary fields. These equations can be written in the symbolic form

$$(I - \Delta t Q_1)u^* = u^n$$
$$(I - \Delta t Q_2)u^{**} = u^* \qquad (11)$$
$$(I - \Delta t Q_3)u^{aux} = u^{**}$$

where I is the identity operator, and Q_ℓ involves differentiations with respect to the variable x_ℓ only. It can be verified that this scheme is accurate to $O(\Delta t)$. If $R \neq 0$, schemes (10) and (11) are accurate to the same order. Scheme (11) is stable in three dimensional problems, while we do not know of a simple extension of (10) to the three dimensional case. Scheme (11) has two useful properties: It requires fewer arithmetic operations per time step than scheme (10), and, because of the simple structure of the right hand sides, the intermediate fields u^*_i, u^{**}_i do not have to be stored separately.

If the scheme (11) is to be used in a problem in which the velocities u_i^n are prescribed at the boundary, values of u_i^*, u_i^{**}, u^{aux} at the boundary have to be provided so that the operators $(I - \Delta t Q_i)$ can be inverted. The scheme is accurate to $O(\Delta t)$, and a variety of choices of u_i^*, u_i^{**}, u_i^{aux} can be made without detracting from that accuracy; for example, we can set

$$u_i^{aux} = u_i^* = u_i^{**} = u_i^n$$

or

$$u_i^{aux} = u_i^* = u_i^{**} = u_i^{n+1}$$

However, in the iteration (4) we need the values of the partial differences of u_i^{aux}, and it can be seen that with a careless choice of u_i^*, u_i^{**}, u_i^{aux} at the wall the error in Du would be $O(\Delta t/\Delta x)$. To ensure that Du^{aux} is accurate to $O(\Delta t)$ we use boundary values of u_i^*, u_i^{**}, u_i^{aux} consistent with the scheme (11), for example

$$\begin{aligned} u_i^* &= u_i^{n+1} - \Delta t Q_2 u_i^{n+1} - \Delta t Q_3 u_i^{n+1} + \Delta t G_i p^n \\ u_i^{**} &= u_i^{n+1} - \Delta t Q_3 u_i^{n+1} + \Delta t G_i p^n \\ u_i^{aux} &= u_i^{n+1} + \Delta t G_i p^n \end{aligned} \tag{12}$$

where $G_i p$ approximates $\partial_i p$ to $O(\Delta t)$. We assume throughout that

$$\Delta t = O(\Delta x_i^2) \ .$$

By inspecting equations (11) we can see that no reasonable results can be expected unless

$$R\Delta t << 1 \ ; \tag{13}$$

this is an eminently reasonable condition; in view of the definitions of R and of the dimensionless time, it merely states that during a given time step, a typical fluid blob covers a distance which is small compared to a typical dimension of the problem. The inequality (13) suggests that as R is increased, the time step Δt has to be decreased.

It should be noted that our non-dimensionalization is not the standard one; our Δt corresponds to a real time interval R timeslonger than the one which corresponds to Δt in the standard non-dimensionalization

$$t' = (\frac{d}{U})t \ .$$

As an illustration, we applied these three schemes to a simple parabolic problem. We solved the heat equation

$$\partial_t u = (\partial_1^2 + \partial_2^2)u$$

in the two dimensional square $\mathcal{S}$: $0 \leq x_1 \leq \pi$, $0 \leq x_2 \leq \pi$, with the initial data

$$u(0) = \sin x_1 \sin x_2$$

and the boundary condition $u = 0$ on the boundary of $\mathcal{S}$. For this problem, $u^{n+1} \equiv u^{aux}$; all three schemes considered: Dufort-Frankel, (10) and (11) are stable for all values of Δt. The Dufort-Frankel scheme however is consistent only when $\Delta t = o(\Delta x_i)$.

The exact solution of the above problem is

$$u(t) = e^{-2t} \sin x_1 \sin x_2$$

In Table I we display the maxima of the differences between the exact solution and the solutions obtained using respectively the Dufort-Frankel scheme and schemes (10) and (11). The Dufort-Frankel scheme requires two levels of initial data; we set

$$u(\Delta t) = e^{-2\Delta t} \sin x_1 \sin x_2$$

in accordance with the exact solution.

We chose $\Delta x_1 = \Delta x_2 = \Delta x = \pi/19$; $\Delta t/\Delta x^2 = 1$. n is the number of time steps. We see that for this value of Δt

Table I: Solutions of the heat equation

n	Dufort-Frankel	A	B
1	---	0.00011	0.00080
2	0.0437	0.00021	0.00152
3	0.0408	0.00030	0.00216
4	0.0314	0.00038	0.00273
5	0.0208	0.00045	0.00323
6	0.0111	0.00051	0.00368
7	0.0028	0.00057	0.00406
8	0.0039	0.00062	0.00440
9	0.0093	0.00066	0.00469
10	0.0134	0.00069	0.00493
20	0.0204	0.00080	0.00574

the Dufort-Frankel scheme is not accurate, and displays oscillations which can be explained by the presence of the term

$$(\frac{\Delta t}{\Delta x})^2 \partial_t^2 u$$

in the expression of the truncation error for that scheme. Scheme (10) is more accurate than scheme (11); we shall see that this advantage is not as notable when the non-linear terms are present in the equations.

The Dufort-Frankel Scheme and successive point over-relaxation.

In order to describe our iteration procedure for determining the pressure and satisfying the equation of continuity, we need a few facts concerning the Dufort-Frankel scheme for the heat equation and its relation to the relaxation method for solving the Laplace equation.

Consider the equation

$$- \nabla^2 u = f, \quad \nabla^2 \equiv \partial_1^2 + \partial_1^2 \tag{14}$$

in some nice domain $\mathcal{D}$, say a rectangle. u is assumed known on the boundary of $\mathcal{D}$. We approximate this equation by

$$- Lu = f \tag{15}$$

where L is the usual five point approximation to the

Laplacian, and u and f are now m-component vectors. m is the number of internal nodes of the resulting difference scheme. For the sake of simplicity we assume that the mesh spacings in the x_1 and x_2 directions are equal, $\Delta x_1 = \Delta x_2 = \Delta x$; this implies no essential restriction. The operator - L is represented by an m x m matrix A which is positive definite, symmetric, and has positive diagonal elements.

We write

$$A = A' - E - E'$$

where E, E' are respectively strictly upper and lower triangular matrices, and A' is diagonal. The convergent relaxation iteration scheme for solving (15) is defined by

$$(A' - \omega E)u^{n+1} = \{(1-\omega)A' + \omega E'\}u^n + \omega f \qquad (16)$$

(see e.g. [5]). ω is the relaxation factor, $0 < \omega < 2$, and the u^n are the successive iterates. The evaluation of the optimal relaxation factor ω_{opt} depends on the fact that A satisfies "Young's condition (A)", i.e. that there exists a permutation matrix P such that

$$P^{-1}AP = \Lambda - N \qquad (17)$$

where Λ is diagonal, and N has the normal form

$$\begin{pmatrix} 0 & G \\ G' & 0 \end{pmatrix}$$

The zero submatrices being square. Under this condition, ω_{opt} can be readily determined (see e.g. [5], Chapter 9).

The matrix A depends on the order in which the components of u^{n+1} are computed from u^n. Changing that order is equivalent to transforming A into $P^{-1}AP$, where P is a permutation matrix.

We now consider the solution of (14) to be the asymptotic steady solution of

$$\partial_\tau u = \nabla^2 u + f \tag{18}$$

and approximate the latter equation by the Dufort-Frankel scheme

$$u^{n+1}_{q,r} - u^{n-1}_{q,r} = \frac{2\Delta\tau}{\Delta x^2}\left(u^n_{q+1,r}+u^n_{q-1,r}+u^n_{q,r+1}+u^n_{q,r-1}-2u^{n+1}_{q,r}-2^{n-1}_{q,r}\right)+2\Delta\tau f$$

$$u^n_{q,r} \equiv u(q\Delta x_1, r\Delta x_2, n\Delta\tau)$$

or, grouping terms

$$\left(1+4\,\frac{\Delta\tau}{\Delta x^2}\right)u^{n+1}_{q,r} - \left(1-4\,\frac{\Delta\tau}{\Delta x^2}\right)u^{n-1}_{q,r}$$

$$\equiv 2\,\frac{\Delta\tau}{\Delta x^2}\left(u^n_{q+1,r}+u^n_{q-1,r}+u^n_{q,r+1}+u^n_{q,r-1}\right) + 2\Delta\tau f$$

Since $u^n_{q,r}$ does not appear in (19), the calculation separates into two independent calculations on intertwined meshes, one of which can be omitted. When this is done, we can write

$$U^{n+1} = \begin{pmatrix} u^{2n} \\ u^{2n+1} \end{pmatrix} \quad (U^{n+1} \text{ has m components})$$

If we then write

$$\omega = \frac{8\,\frac{\Delta\tau}{\Delta x^2}}{1+4\,\frac{\Delta\tau}{\Delta x^2}} \tag{20}$$

we see that the iteration (19) reduces to an iteration of the form (16) where the new components of U^{n+1} are calculated in an order such that A has the normal form (17). The Dufort-Frankel scheme appears therefore to be a particular ordering of the overrelaxation method whose existence is equivalent to Young's condition (A).

The best value of $\Delta\tau$, $\Delta\tau_{opt}$, can be determined from ω_{opt} and relation (20). We find that $\Delta\tau_{opt} = O(\Delta x)$, and therefore for $\Delta\tau = \Delta\tau_{opt}$ the Dufort-Frankel scheme approximates, not equation (18), but rather the equation

$$\partial_\tau u = \nabla^2 u - 2(\frac{\Delta\tau}{\Delta x})^2\, \partial_\tau^2 u \ .$$

This is the equation which Garabedian in [6] used to estimate ω_{opt}. It can be used here to estimate $\Delta\tau_{opt}$.

These remarks obviously generalize to problems where $\Delta x_1 \neq \Delta x_2$ or where there are more than two space variables.

The iteration procedure for determining u_i^{n+1}, p^{n+1}.

For the sake of clarity we shall assume in this section that the domain $\mathcal{D}$ is two dimensional and rectangular, and that the velocities are prescribed at the boundary. Extension of the procedure to three dimensional problems is immediate, and extension to problems with other types of boundary conditions often possible. Stress-free boundaries and periodicity conditions in particular offer no difficulty.

Let $\mathcal{B}$ denote the boundary of $\mathcal{D}$ and $\mathcal{C}$ the set of mesh nodes with a neighbor in $\mathcal{B}$. In $\mathcal{D} - \mathcal{B}$ we approximate the equation of continuity by centered differences, i.e. we set

$$Du \equiv \frac{1}{2\Delta x_1}\left(u_{1(q+1,r)} - u_{1(q-1,r)}\right) + \frac{1}{2\Delta x_2}\left(u_2(q,r+1) - u_{2(q,r-1)}\right) = 0 \tag{21}$$

At the points of $\mathcal{B}$ we use second order one-sided differences, so that Du is accurate to $O(\Delta x^2)$ everywhere. Consider the boundary line $x_2 = 0$, represented by $j = 1$ (Fig. 1). We have on that line

$$Du \equiv \frac{2}{\Delta x_2}\left[u_{2(q,2)} - u_{2(q,1)} - \frac{1}{4}\left(u_{2(q,3)} - u_{2(q,1)}\right)\right] + \frac{1}{2\Delta x_1}\left(u_{1(q+1,1)} - u_{1(q-1,1)}\right) = 0 \tag{22}$$

with similar expressions at the other boundaries.

Let (q,r) be a node in $\mathcal{D} - \mathcal{B} - \mathcal{C}$; knowing $u_i^{n+1,m}$, $i = 1,2$, and $p^{n+1,m}$, we shall evaluate simultaneously

$u_{1(q\pm1,r)}^{n+1,m+1}$, $u_{2(q,r\pm1)}^{n+1,m+1}$ and $p_{q,r}^{n+1,m}$ ($p_{q,r} \equiv p(q\Delta x_1, r\Delta x_2)$), using the formulae

$$p_{q,r}^{n+1,m+1} = p_{q,r}^{n,m} - \lambda D u^{n+1,m+1} \tag{23a}$$

(D given by (21))

$$u_{1(q+1,r)}^{n+1,m+1} = u_{1(q+1,r)}^{aux} - \frac{\Delta t}{2\Delta x_1}\left(p_{q+2,r}^{n+1,m} - \frac{1}{2}\left(p_{q,r}^{n+1,m+1} - p_{q,r}^{n+1,m}\right)\right) \tag{23b}$$

$$u_{1(q-1,r)}^{n+1,m+1} = u_{1(q-1,r)}^{aux} - \frac{\Delta t}{2\Delta x_1}\left(\frac{1}{2}\left(p_{q,r}^{n+1,m+1} + p_{q,r}^{n+1,m}\right) - p_{q-2,r}^{n+1,m}\right) \tag{23c}$$

$$u_{2(q,r+1)}^{n+1,m+1} = u_{2(q,r+1)}^{aux} - \frac{\Delta t}{2\Delta x_2}\left(p_{q,r+2}^{n+1,m} - \frac{1}{2}\left(p_{q,r}^{n+1,m+1} + p_{q,r}^{n+1,m}\right)\right) \tag{23d}$$

$$u_{2(q,r-1)}^{n+1,m+1} = u_{2(q,r-1)}^{aux} - \frac{\Delta t}{2\Delta x_2}\left(\frac{1}{2}\left(p_{q,r}^{n+1,m+1} + p_{q,r}^{n+1,m}\right) - p_{q,r-2}^{n+1,m}\right) \tag{23e}$$

(See Fig. 2).

These equations specify $G^m p$ for use in equation (4b).

In $\mathcal{C}$ and $\mathcal{D}$ these formulae have to be modified. Consider again the boundary line $x_2 = 0$ (Fig. 1). Assume the velocities are prescribed at the boundary, i.e. $u_{i(q,1)}^{n+1}$ are given, $i = 1, 2$. There are several ways of including that information in the iterations (4). The consistent way (in the sense of equations (12)) would be to set

$$u_{i(q,1)}^{aux} = u_{i(q,1)}^{n+1} + \Delta t G_i p^n$$

and

$$u_{i(q,1)}^{n+1,m+1} = u_{i(q,1)}^{n+1}$$

This is the better approach, which we intend to use in future work. However, for the sake of simplicity, we chose here an inconsistent way of treating the boundary, i.e. we set

$$u^{aux}_{i(q,1)} = u^{m+1,m}_{i(q,1)} = u^{n+1,m+1}_{i(q,1)} = u^{n+1}_{i(q,1)} \tag{24}$$

It can easily be verified that this does not affect the values of u_i^{n+1}; it does however introduce an additional error of $O(\Delta t)$ into the computed pressure field.

Equations (23) and their modifications near the boundary can be solved for $p^{n+1,m+1}$, yielding, for (q,r) in $\mathcal{D} - \mathcal{B} - \mathcal{C}$:

$$\begin{aligned} p^{n+1,m+1}_{q,r} = (1 + \alpha_1 + \alpha_2)^{-1}[&(1 - \alpha_1 - \alpha_2)p^{n+1,m} - \lambda Du^{aux} \\ &+ \alpha_1(p^{n+1,m}_{q+2,r} + p^{n+1,m}_{q-2,r}) + \alpha_2(p^{n+1,m}_{q,r+2} + p^{n+1,m}_{q,r-2})] , \end{aligned} \tag{25a}$$

for (q,r) in $\mathcal{C}$.

$$\begin{aligned} p^{n+1,m+1}_{q,2} = (1 + \alpha_1 + \tfrac{1}{2}\alpha_2)^{-1}\ [&(1 - \alpha_1 - \tfrac{1}{2}\alpha_2)p^{n+1,m}_{q,2} - \lambda Du^{aux} \\ &+ \alpha_1(p^{n+1,m}_{q+2,2} + p^{n+1,m}_{q-2,2}) + \alpha_2 p^{n+1,m}_{q,4}] , \end{aligned} \tag{25b}$$

and for (q,r) on $\mathcal{B}$:

$$\begin{aligned} p^{n+1,m+1}_{q,1} = (1 + \alpha_1)^{-1}[&(1 - \alpha_1)p^{n+1,m}_{q,1} - \lambda Du^{aux} \\ &+ 2\alpha_2(p^{n+1,m}_{q,3} - \tfrac{1}{4}(p^{n+1,m}_{q,4} - p^{n+1,m}_{q,2}))] \end{aligned} \tag{25c}$$

etc. In (25a) and (25b) Du is given by (21), and in (25c) by (22). The several representations of Du have been used to derive these expressions, and we have written

$$\alpha_i = \frac{\lambda \Delta t}{4 \Delta x_i^2} \qquad i = 1, 2 .$$

These iterations are to be performed until for some ℓ

$$\max_{q,r} |p_{q,r}^{n+1,\ell+1} - p_{q,r}^{n+1,\ell}| \leq \varepsilon$$

for a predetermined ε.

Formula (25a) is seen to be a Dufort-Frankel relaxation scheme for the solution of an analogue of equation (8); this of course was to be expected. The $\Delta\tau$ of the preceding section is replaced here by $\lambda\Delta t/2$; corresponding to $\Delta\tau_{opt}$ (or ω_{opt}) we find λ_{opt}. If p were known on $\mathscr{B}$ or $\mathscr{C}$ and the iterations (25a) stood alone, convergence of the iteration for all $\lambda > 0$ would follow from the argument of the preceding section, and $\lambda = \lambda_{opt}$ would lead to the fastest convergence. Although no proof is offered, the numerical evidence leads us to state that the whole iteration system--equations (25a), (25b), (25c)--converges for all $\lambda > 0$, and converges fastest when $\lambda = \lambda_{opt}$. None of the boundary instabilities reported by users of the stream function vorticity approach has been observed.

In (25a) the Laplacian of p is evaluated by a difference formula using a stencil whose nodes are separated by $2\Delta x_1$, $2\Delta x_2$. This results from the use of centered differences for approximating both $\partial_j u_j$ and $\partial_i p$. As a consequence, the pressure iterations split into two calculations on intertwined

meshes, coupled at the boundary. The most efficient orderings for performing the iterations (25) are such that the resulting over-all scheme is a Dufort-Frankel scheme for each one of the intertwined meshes. This involves no particular difficulty; a possible ordering for a rectangular grid is shown in Fig. 3.

The new velocities u_i^{n+1}, i = 1, 2, are to be evaluated using (23a), (23b), (23c), and (23d). This has to be done only after the $p^{n+1,m}$ have converged; there is no need to evaluate and store the intermediate fields $u_i^{n+1,m+1}$. A saving in computing time can be made by evaluating Du^{aux} once and for all at the beginning of the iteration.
An advantage of our formulation appears here: The iterates $p^{n+1,m}$ in (25a), (25b), (25c) converge to the solution of a certain set of equations. Suppose we were to solve this set of equations by some other iteration procedure, stopping when two successive iterates $p^{n+1,\ell+1}$, $p^{n+1,\ell}$ differ by less than a small number ε; we could then use the latest iterate, $p^{n+1,\ell+1}$ to evaluate u_i^{n+1}, though formulae such as

$$u_i^{n+1} = u_i^{aux} - G_i p^{n+1,\ell+1}$$

where $G_i p$ approximates $\partial_i p$. If $\Delta t = O(\Delta x^2)$, Du^{n+1} would fail to vanish by an amount $O(\varepsilon)$. On the other hand, using formulae (23), when $p^{n+1,\ell+1}$ and $p^{n+1,\ell}$ differ by less than ε, $Du^{n+1} = O(\varepsilon/\lambda)$; it can be seen that $\lambda_{opt} = O(\Delta x^{-1})$,

therefore $Du^{n+1} = O(\varepsilon\Delta x)$; a gain in accuracy appears, which can be used to relax the convergence criterion for the iterations.

Solution of a test problem. The proposed method was first applied to a simple two dimensional test problem, used as a test problem by Pearson [7] for a vorticity-stream function method. $\mathscr{D}$ is the square $0 \leq x_i \leq \pi$, $i = 1, 2$; $E_1 = E_2 = 0$; the boundary data are

$$u_1 = - \cos x_1 \sin x_2 \, e^{-2t}$$

$$u_2 = \sin x_1 \cos x_2 \, e^{-2t}$$

The initial data are

$$u_1 = - \cos x_1 \sin x_2$$

$$u_2 = \sin x_1 \cos x_2$$

The exact solution of the problem is

$$u_1 = - \cos x_1 \sin x_2 \, e^{-2t}$$

$$u_2 = \sin x_1 \cos x_2 \, e^{-2t}$$

$$p = - R \, \frac{1}{4} (\cos 2x_1 + \cos 2x_2) e^{-4t}$$

where R is the Reynolds number. We assume that

$$\Delta x_1 = \Delta x_2 = \Delta x$$

λ_{opt} is then found to be

$$\lambda_{opt} = \frac{2\Delta x^2}{\Delta t \sin(2\Delta x)}$$

In tables II, III, IV, V and VI we display the results of some sample calculations. n is the number of time steps; $e(u_i)$, i = 1, 2, are the maxima over $\mathcal{D}$ of the difference between the exact and the computed solutions u_i. It is not clear how the error in the pressure is to be represented; p^n is defined at a time intermediate between $(n-1)\Delta t$ and $n\Delta t$; it is proportional to R in our non-dimensionalization. There are errors in p due to the use of the inconsistent boundary data (24), and due also to the fact, discussed at the end of the preceding section, that the iterations can be stopped before the $p^{n,m}$ have truly converged. e(p) in the tables represents the maximum over the grid of the differences between the exact pressure at time $n\Delta t$ and the computed p^n, divided by R; it is given mainly for the sake of completeness. The accuracy of the scheme is to be judged by the smallness of $e(u_i)$. ℓ is the number of iterations; it is to be noted that the first iteration has to be performed in order to approximate equation (1). "Scheme A" means formulae (10) were used to evaluate u^{aux}, "Scheme B" means formulae (11) were used. ε is the convergence criterion.

Table II

Scheme A; $\Delta x = \pi/39$; $\Delta t = 2\Delta x^2 = 0.01397$; $\varepsilon = \Delta x^2$; R = 1

n	$e(u_1)$	$e(u_2)$	e(p)	ℓ
1	2.8×10^{-4}	2.6×10^{-4}	0.0243	1
2	2.7×10^{-4}	2.0×10^{-4}	0.0136	7
3	1.5×10^{-4}	1.3×10^{-4}	0.0069	4
4	1.8×10^{-4}	1.9×10^{-4}	0.0145	4
5	1.3×10^{-4}	1.7×10^{-4}	0.0089	5
6	1.3×10^{-4}	1.8×10^{-4}	0.0116	4
7	1.6×10^{-4}	1.9×10^{-4}	0.0144	4
9	1.4×10^{-4}	1.7×10^{-4}	0.0147	4
10	1.3×10^{-4}	1.6×10^{-4}	0.0156	4
20	1.8×10^{-4}	2.3×10^{-4}	0.0241	4

Table III

Scheme A; $\Delta x = \pi/39$; $\Delta t = 2\Delta x^2 = 0.01397$; $\varepsilon = \Delta x^3$; $R = 1$

n	$e(u_1)$	$e(u_2)$	$e(p)$	ℓ
1	8.5×10^{-5}	3.8×10^{-5}	0.0059	10
2	1.0×10^{-4}	5.7×10^{-5}	0.0067	10
3	1.0×10^{-4}	7.0×10^{-5}	0.0068	10
4	1.0×10^{-4}	7.8×10^{-5}	0.0068	10
5	1.0×10^{-4}	8.3×10^{-5}	0.0069	10
6	9.7×10^{-5}	8.6×10^{-5}	0.0070	10
7	9.4×10^{-5}	8.7×10^{-5}	0.0071	10
8	9.0×10^{-5}	8.7×10^{-5}	0.0073	10
9	8.7×10^{-5}	8.7×10^{-5}	0.0077	10
10	8.3×10^{-5}	8.5×10^{-5}	0.0082	10
20	1.0×10^{-4}	1.0×10^{-4}	0.0216	9

Table IV

Scheme A; $\Delta x = \pi/39$; $\Delta t = \Delta x^2 = 0.00648$; $\varepsilon = \Delta x^2$; R = 20

n	$e(u_1)$	$e(u_2)$	e(p)	ℓ
1	2.5×10^{-3}	2.7×10^{-3}	0.0167	15
3	3.7×10^{-3}	3.5×10^{-3}	0.0175	10
5	5.5×10^{-3}	4.5×10^{-3}	0.0183	10
7	6.8×10^{-3}	5.2×10^{-3}	0.0189	10
9	7.5×10^{-3}	5.4×10^{-3}	0.0193	10
20	7.0×10^{-3}	4.7×10^{-3}	0.0204	9

Table V

Scheme B; $\Delta x = \pi/39$; $\Delta t = \Delta x^2 = 0.00648$; $\varepsilon = \Delta x^2$; R = 20

n	$e(u_1)$	$e(u_2)$	e(p)	ℓ
1	3.9×10^{-3}	4.4×10^{-3}	0.0404	16
3	5.9×10^{-3}	6.0×10^{-3}	0.0466	11
5	8.5×10^{-3}	6.7×10^{-3}	0.0505	10
7	1.0×10^{-2}	7.4×10^{-3}	0.0551	10
9	1.1×10^{-2}	7.9×10^{-3}	0.0599	10
20	1.0×10^{-2}	7.8×10^{-3}	0.0839	10

Table VI

Scheme B; $\Delta x = \pi/39$; $\Delta t = \frac{1}{2}\Delta x^2 = 0.00324$; $\varepsilon = \Delta x^2$; R = 20

n	$e(u_1)$	$e(u_2)$	e(p)	ℓ
1	1.1×10^{-3}	1.2×10^{-3}	0.0217	15
3	1.9×10^{-3}	2.1×10^{-3}	0.0234	9
5	2.5×10^{-3}	2.8×10^{-3}	0.0242	9
7	3.3×10^{-3}	3.2×10^{-3}	0.0249	9
9	4.0×10^{-3}	3.5×10^{-3}	0.0253	8
20	5.8×10^{-3}	3.9×10^{-3}	0.0258	8

Tables II and III describe computations which differ only in the value of ε. They show that $\varepsilon = \Delta x^2$ is an adequate convergence criterion. Tables IV and V indicate that reliable results can be obtained even when the Reynolds number is large for the grid employed; when $R = 20$, $\Delta x = \frac{\pi}{39}$, we have

$$R \simeq 1.5\ \Delta x^{-1}$$

The errors are of the order of 1 %. Tables IV and V show that scheme B does not introduce unduly large errors into the calculations; table VI shows that, as expected, a decrease in Δt decreases the error (but increases the amount of computational labor).

Application to thermal convection. Suppose a plane layer of fluid, of thickness d and infinite lateral extent, is heated from below. The lower boundary $x_3 = 0$ is maintained at a temperature T_o, the upper boundary $x_3 = d$ at a temperature $T_1 < T_o$. The warmer fluid at the bottom expands and tends to move upward; this motion is inhibited by the viscous stresses.

The equations describing the fluid motion are, in the Boussinesq approximation (see e.g. [8], Chapter 2)

$$\partial_t u_i + u_j \partial_j u_i = -\frac{1}{\rho_o}\partial_i p + \nu\nabla^2 u_i - g(1 - (T - T_o))\delta_i$$

$$\partial_t T + u_j \partial_j T = k\nabla^2 T \qquad \partial_j u_j = 0$$

where T is the temperature, k the coefficient of thermal conductivity, α the coefficient of thermal expansion, and δ_i the components of the unit vector pointing upwards.

We write

$$u_i' = (\frac{d}{\nu})u_i \qquad T' = \frac{T-T_1}{T_o-T_1} \qquad t' = (\frac{\nu^2}{d})t$$

$$x_i' = \frac{x_i}{d}, \qquad p' = \frac{1}{\rho_o}(\frac{d}{\nu})^2 p + \frac{(T_1-T_o)dgx_3}{\nu^2}$$

and drop the primes. The equations now are

$$\partial_t u_i + u_j \partial_j u_i = -\partial_i p + \nabla^2 u_i + \frac{R}{\sigma}(T-1)\delta_i$$

$$\partial_t T + u_j \partial_j T = \frac{1}{\sigma}\nabla^2 T$$

$$\partial_j u_j = 0$$

where $R^* = \alpha g d^3(T_o - T_1)/k\nu$ is the Rayleigh number, and $\sigma = \nu/k$ the Prandtl number. The rigid boundaries are now situated at $x_3 = 0$ and $x_3 = 1$, where it is assumed that $u_i = 0$, $i = 1, 2, 3$.

It is known that for $R^* < R_c^*$, the state of rest is stable and no steady convection can arise, with $R_c^* = 1707.762$ (see [9]). When $R^* = R_c^*$, steady infinitesimal convection

can first appear, and the field quantities are given by

$$u_3 = CW(x_3)\phi$$
$$u_i = \frac{C}{a^2} W(x_3)\partial_i\phi \quad i = 1, 2$$
$$T = CT(x_3)\phi$$

where $\phi = \phi(x_1,x_2)$ determines the horizontal planform of the motion and satisfies

$$(\partial_1^2 + \partial_2^2 + a^2)\phi = 0 ,$$

$W(x_3),T(x_3)$ are fully determined functions of x_3, $a = 3.117$, and C is a small but undetermined amplitude. (see [8]).

In two dimensional motion $u_1 = 0$ and the motion does not depend on x_1. We then have

$$\phi = \cos ax_2$$

The motion is periodic in x_2 with period $2\pi/a$.

The Nusselt number Nu is defined as the ratio of the total heat transfer to the heat transfer which would have occured in no convection were present. For $R^* \leq R_c^*$, Nu = 1. In our dimensionless variables

$$Nu = \frac{a}{2\pi}\int_0^{2\pi/a} (\sigma u_3 T - \partial_3 T)dx_2$$

In [1] and [10] we studied the dependence of Nu on R* and σ for steady convection, assuming that the motion remains periodic in x_2 with period $2\pi/a$. There seems to be no point

in reproducing these calculations with the present method; some values of Nu were recomputed and are in good agreement with those obtained previously.

It appears of interest to determine the onset of instability by finite differences. We assume that the flow is periodic in x_2 with period $2\pi/a$, and we define $r = R^*/R_c^*$. We find that for $r = 1$, Nu oscillates between 1 and 1.03 for a very long time; for $r = 0.99$, Nu = 1.0000; i.e. the computed and the exact values of R_c^* agree to within less than 1 %. These results were obtained with $\Delta x_2 = 2\pi/28a$, $\Delta x_3 = 1/27$, $\varepsilon = \Delta x_2^2$, $\Delta t = 3\Delta x_3^2$. The corresponding λ_{opt} is

$$\lambda_{opt} = \frac{4}{(\frac{\Delta t}{\Delta x_1^2} + \frac{\Delta t}{\Delta x_2^2})\sqrt{1-\rho^2}}, \quad \rho = 1 - \frac{2\Delta x_2^2 \sin^2(\pi \Delta x_3)}{(\Delta x_2^2 + \Delta x_3^2)}$$

In three dimensional convection problems not only the amplitude of the motions is to be determined but also their spatial configuration. For $R^* = R_c^*$, ϕ can be any periodic solution of

$$(\partial_1^2 + \partial_2^2 + a^2)\phi = 0 \qquad a = 3.117$$

It is reasonable to assume that the cell patterns are made up of polygons whose union covers the (x_1, x_2) plane. Possible cell shapes are hexagons, squares, and rolls (i.e. two dimensional convection cells). It is known (see e.g. [11]) that the equations of motion admit steady

solutions corresponding to more than one kind of cell. However, only cellular structures which are stable with respect to small perturbations can persist in nature or be exhibited by our method. It has been shown, numerically by the author [10], experimentally by Koschmieder [12] and Rossby [13], theoretically, in the case of infinite σ and small perturbations, by Busse [14], that for $R^*/R_c^* < 10$ the stable cellular mode is a roll with a wave number in a certain range. We shall now demonstrate numerically the impermanence of hexagonal convection and the emergence of a roll.

We choose $R^*/R_c^* = r = 2$, $\sigma = 1$. We assume the motion to be periodic in the x_1 and x_2 directions, with periods respectively $4\pi//\sqrt{3}a$ and $4\pi/a$ (the first period is apparently in the stable range of periods as predicted by Busse [14]). These are the periods of the hexagonal cells,which could arise when $R^* = R_c^*$. The state of rest is perturbed by adding to the temperature in the plane $x_3 = \Delta x_3$ a multiple of the function $\phi(x_1,x_2)$ which corresponds to a hexagonal cell, and adding a small constant to the temperature on the line $x_1 = \frac{3}{4}(4\pi/\sqrt{3}a)$, $x_2 = \frac{3}{4}(4\pi/a)$. We then follow the evolution of the convection in time. We use a net of 24 x 24 x 25, i.e.

$$\Delta x_1 = (4\pi/\sqrt{3}a)/24 ,$$

$$\Delta x_2 = (4\pi/a)/24$$

$$\Delta x_3 = 1/24$$

we also set $\varepsilon = \Delta x_2^2$

$$\Delta t = 3\Delta x_3^2$$

We then find

$$\lambda_{opt} = \frac{4}{(\frac{\Delta t}{\Delta x_1^2} + \frac{\Delta t}{\Delta x_2^2} + \frac{\Delta t}{\Delta x_3^2})\sqrt{1-\rho^2}}$$

where $$\rho = 1 - 2\frac{\Delta x_1^2 \Delta x_2^2 \sin^2 \pi \Delta x_3}{(\Delta x_1^2 \Delta x_2^2 + \Delta x_1^2 \Delta x_3^2 + \Delta x_2^2 \Delta x_3^2)}$$

To visualize the convection pattern, we proceed as follows: We consider the plane $x_3 = 17\Delta x_3$; if $u_{3(q,r,18)} > 0$ we print an *, if $u_{3(q,r,18)} \leq 0$ we print a 0.

The evolution of the convection is shown in Fig. 4a, 4b, 4c, 4d, 4e and 4f. The hexagonal pattern introduced into the cell is not preserved. The system first attempts to achieve equilibrium as a roll with period $4\pi/a$ (Fig. 4c), then as a roll with period $8\pi/\sqrt{3}a$ (Fig. 4d), and finally becomes a roll with period $4\pi/\sqrt{3}a$ (Fig. 4f).

With different initial perturbations the evolution of the system is different, the final result is the same. The calculation was not pursued until steady convection had been established, because that would have been excessively time consuming. It is known from previous studies that steady rolls can be obtained. (See e.g. [1]), and that the mesh

used here provides an adequate representation.

Conclusion and applications. The Benard convection problem is not considered to be an easy problem to solve numerically even in the two dimensional case (see e.g. [15]). The fact that with our method reliable time dependent results can be obtained even in three space dimensions indicates that the Navier-Stokes equations do indeed lend themselves to numerical solution. A number of applications to convection problems, with or without rotation, can be contemplated; in particular, it appears to be of interest to study systematically the stability of Benard convection cells when $\sigma \neq \infty$, and when the perturbations have a finite amplitude. The existence of finite amplitude instability of the cells is suggested by the numerical results reported in [10].

Other applications should include the study of the finite amplitude instability of Poiseuille flow, the stability of Couette flow, and similar problems.

Another type of application, presently being carried out by the author, includes the finite difference solution of the Green-Taylor problem [16] and related problems involving simplified representations of turbulence.

Acknowledgements. The author would like to thank Professors Peter D. Lax and Herbert B. Keller for their interest and for helpful discussions and comments.

Bibliography

[1] A. J. Chorin, J. of Computational Physics, 2, 1, (1967).

[2] F. H. Harlow and J. E. Welch, The Physics of Fluids, 8, 2182, (1965).

[3] J. O. Wilkes, Ph.D. Thesis, U. of Michigan, (1963).

[4] A. A. Samarskii, U.S.S.R. Computational Mathematics and Mathematical Physics, 5, 894, (1963).

[5] R. Varga, Matrix Iterative Analysis, Prentice Hall (1962).

[6] P. R. Garabedian, Math. Tables Aids. Comput., 10, 183, (1956).

[7] C. E. Pearson, Sperry-Rand Research Center, Sudbury, Mass., Report No. SRRC-RR-64-17, (1964).

[8] S. Chandrasekhar, Hydrodynamic and Hydromagnetic Stability, Oxford, the Clarendon Press (1961).

[9] D. D. Joseph, Arch. Rational Mech. Anal., 20, 59, (1965).

[10] A. J. Chorin, Report No. NYO-1480-61, New York University, (1966).

[11] P. H. Rabinowitz, to appear.

[12] E. L. Koschmieder, Beitr. Physik. Atm. 39, 1, (1966).

[13] H. T. Rossby, Ph.D. Thesis, M.I.T., (1966).

[14] F. Busse, J. Math. Phys. 46, 140, (1967).

[15] P. Schneck and G. Veronis, Physics of Fluids, 10, 927, (1967).

[16] G. I. Taylor and A. E. Green, Proc. Roy. Soc. A, 158, 499, (1937).

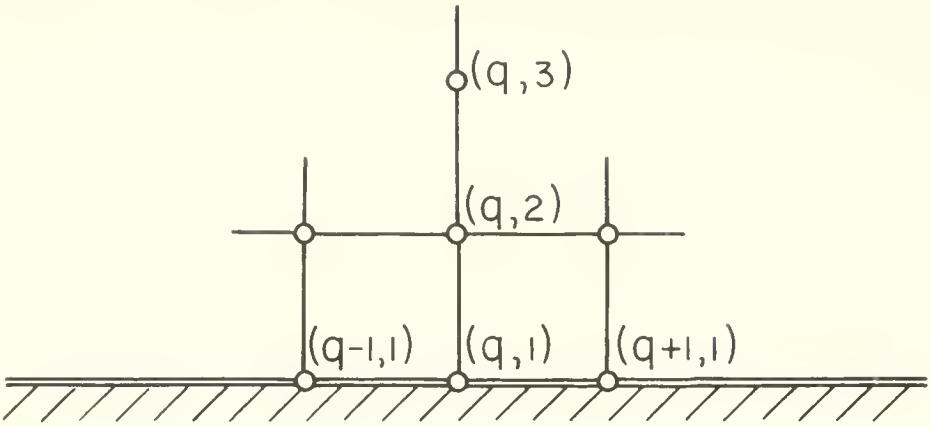


Figure 1. Mesh Near a Boundary

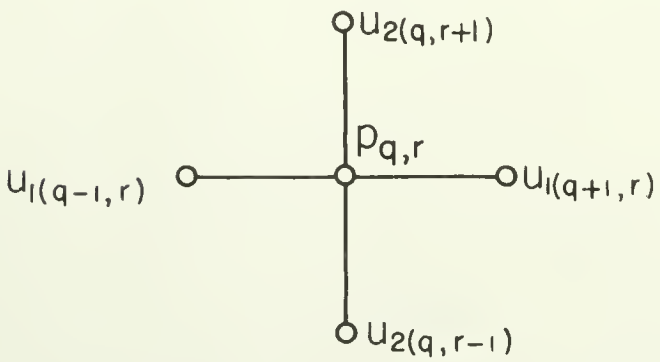
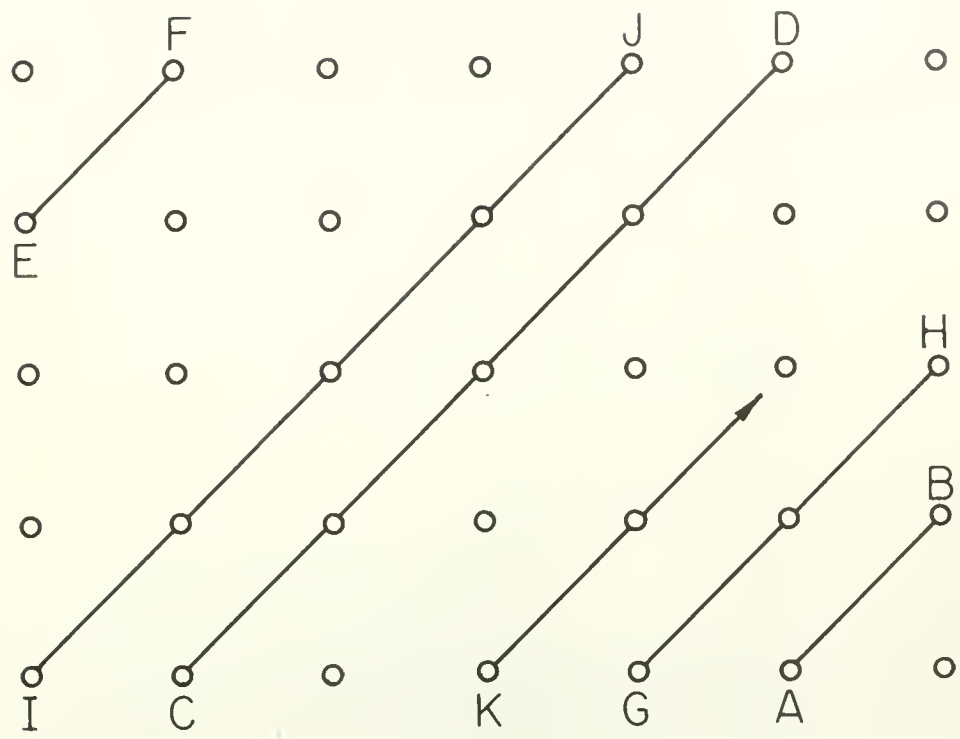


Figure 2. Iteration Scheme



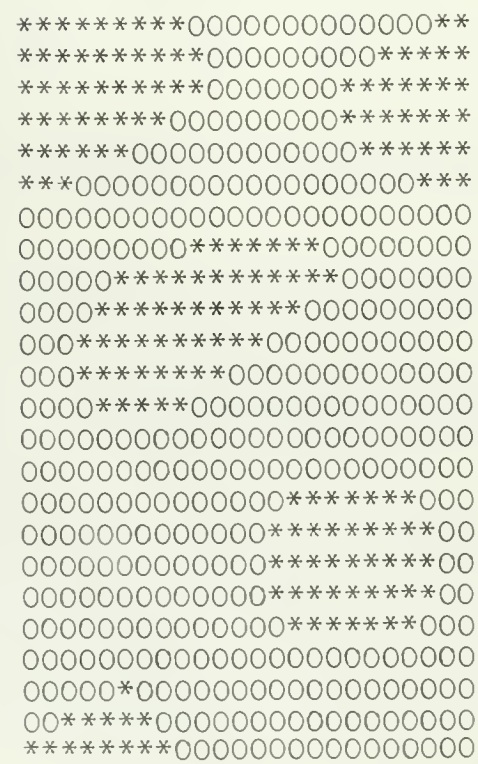
The domain is sweeped in the order AB, CD, EF, GH, IJ, K

Figure 3. An Ordering for the Iteration Scheme

Fig. 4. Evolution of a Convection Cell

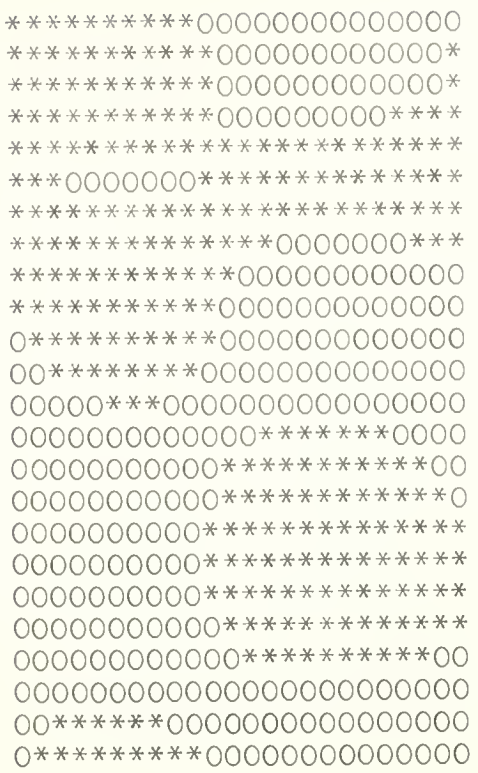
Nu = 1.000

4a. After 1 step



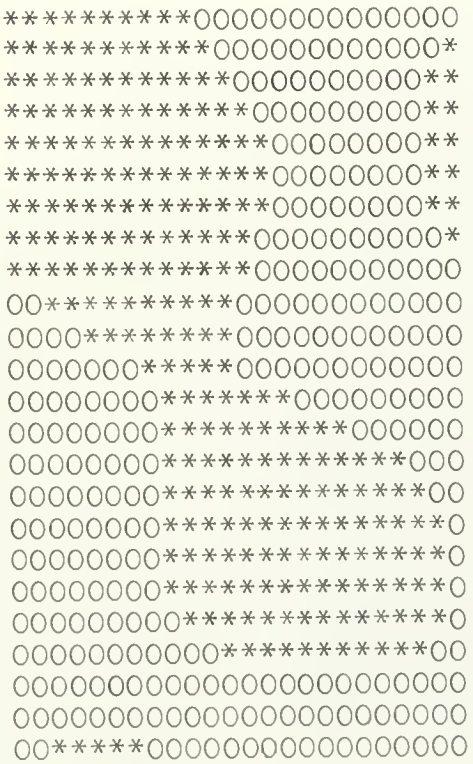
Nu = 1.000

4b. After 10 steps



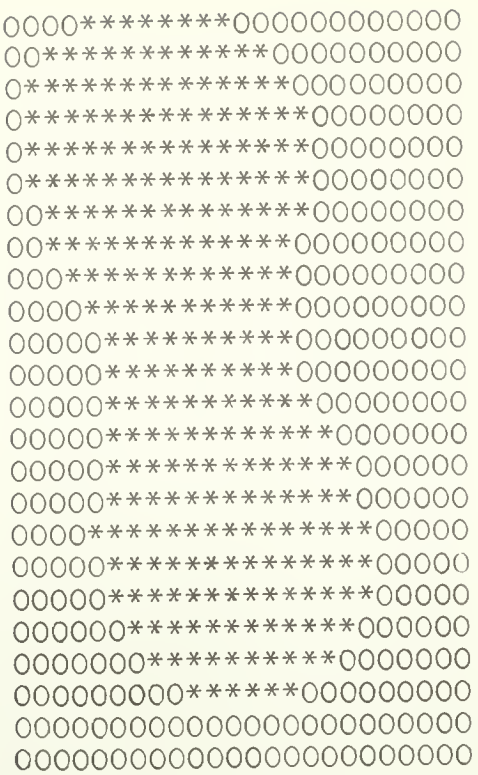
Nu = 1.25

4c. After 125 steps



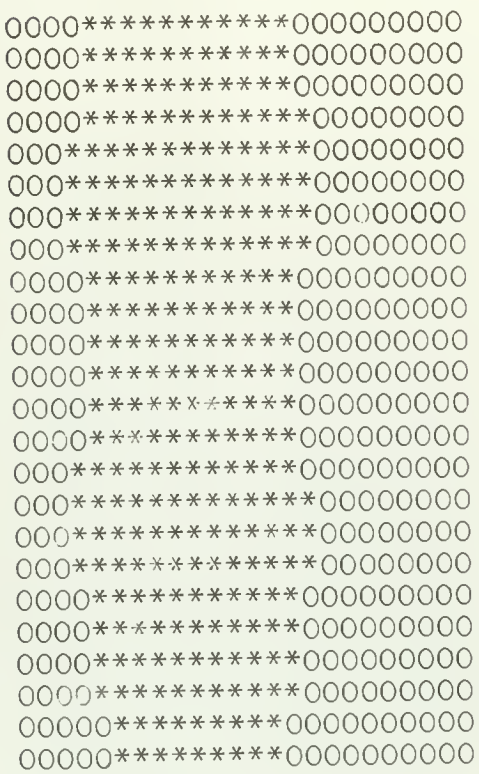
Nu = 1.72

4d. After 225 steps



Nu = 1.76

4e. After 325 steps



Nu = 1.77

4f. After 430 steps